Analyzing and overcoming the effects of GNSS error on LiDAR based orchard parameters estimation

Darío Javier Guevara[a], Fernando A. Auat Cheein*[a], Jordi Gené-Mola[b], Joan Ramón Rosell-Polo[b], Eduard Gregorio[b]

[a] *Department of Electronic Engineering, Universidad Técnica Federico Santa María, Valparaíso, Chile.*
[b] *Department of Agricultural and Forest Engineering, University of Lleida, Spain.*

Abstract

Currently, 3D point clouds are obtained via LiDAR (Light Detection and Ranging) sensors to compute vegetation parameters to enhance agricultural operations. However, such a point cloud is intrinsically dependent on the GNSS (global navigation satellite system) antenna used to have absolute positioning of the sensor within the grove. Therefore, the error associated with the GNSS receiver is propagated to the LiDAR readings and, thus, to the crown or orchard parameters. In this work, we first describe the error propagation of GNSS over the laser scan measurements. Second, we present our proposal to overcome this effect based only on the LiDAR readings. Such a proposal uses a scan matching approach to reduce the error associated with the GNSS receiver. To accomplish such purpose, we fuse the information from the scan matching estimations with the GNSS measurements. In the experiments, we statistically analyze the dependence of the grove parameters extracted from the 3D point cloud -specifically crown surface area, crown vol-

Email address: fernando.auat@usm.cl (Fernando A. Auat Cheein*)

ume, and crown porosity- to the localization error. We carried out 150 trials with positioning errors ranging from 0.01 meters (ground truth) to 2 meters. The results when only GNSS was used as a localization system showed that errors associated with the estimation of vegetation parameters increased more than 100 % when GNSS positioning error was equal or bigger than 1 meter. On the other hand, when our proposal was used as a localization system, the results showed that for the same case of 1 meter, the estimation of orchard parameters improved in 20 % overall. However, in lower positioning errors of the GNSS, the estimation of orchard parameters were improved up to 50% overall. These results suggest that our work could lead to better decisions in agricultural operations, which are based on foliar parameter measurements, without the use of external hardware.

Keywords:

Global positioning system, point cloud registration, mobile sensing, vehicle localization, phenotyping.

---

1. Introduction

Precision agriculture is benefited from the use of communication and information technologies, for enhancing the decision-making process and for automation of a wide range of agricultural tasks. In particular, the use of new sensors and sensing technologies improved the capabilities of the farmer to take preventive actions, such as in the case of weed monitoring and plague detection (Underwood et al. 2017).

Among the new sensing technologies, LiDAR (light detection and range) sensors are used mainly for geometric characterization of groves (Rilling et al.

2017; Trochta et al. 2017; Rosell and Sanz 2012; Li et al. 2018). From the geometric parameters of plants, the crown surface area and volume are of special interest since both combine the width, height, geometric shape and the structure of trees (Auat Cheein et al. 2015). Such parameters are commonly used for farmers for, e.g., stablish a biomass model for plants (Lin et al. 2017), herbicide management and pruning directives (Rosell et al. 2009).

Geometric parameters estimate from raw data are mainly affected by the uncertainties in the laser scanner, calibration of the setup and the georeferencing system. Different works have analyzed the error propagation in scanner laser systems (Mezian et al. 2016; Goulden and Hopkinson 2010; Hartzell et al. 2015). However, those analyses do not consider the GNSS positioning error. Similarly, in the analysis of tree volume sensitivity developed in (Palleja et al. 2010), the GNSS positioning error is not considered. Such work considers specific errors such as vehicle speed, the height of the LiDAR, distance of the measurement and the angular orientation of the LiDAR.

As shown in the references mentioned above, GNSS positioning error is not considered when building the LiDAR-based point cloud. This fact occurs although the magnitude of the navigation system positioning errors has a direct impact on the resultant point cloud, which is not dependent upon any of the other uncertainties (Glennie 2008). Instead, the point clouds merging relies on the ability of the programmer (Rosell and Sanz 2012; Rosell et al. 2009) or on commercial software (Sanz et al. 2018) where no considerations are taken into account regarding GNSS absolute errors. Therefore, error propagation, from the GNSS antenna to every point from the point cloud, is disregarded.

Commonly, LiDAR-based point cloud information (Gaulton and Malthus 2010; St-Onge et al. 2008) is used for mapping approaches. However, when the LiDAR is mounted on a vehicle, such information can also be used in scan matching techniques for vehicle localization (Grant et al. 2019; Malavazi et al. 2018). The goal of those approaches is to find the position of the sensor over time by finding the rigid transformation matrix which maximizes the overlap between point cloud scans (2D or 3D) obtained at different times. The process of finding the rigid transformation matrix is known as *registration*.

There are several variations of registration algorithms, for example, the Normal Distribution Transform (Magnusson et al. 2009), which creates a voxel grid over the point clouds or the Gaussian Mixture Model (Boughorbel et al. 2004), which performs registration based on expectation maximization. Other algorithms focuses on specific features from the scene to perform registration, such as corners or planes (Lamine Tazir et al. 2018; Peng et al. 2016). A well-known alternative is the iterative closest point registration, which has been widely used in the last years (Ren et al. 2019; Kim et al. 2018; Donoso et al. 2017); therefore, many variations have been proposed, such as EM-ICP (Granger and Pennec 2002) and Generalized-ICP (Segal et al. 2009). All these techniques can provide the pose of a vehicle when the LiDAR is mounted on it. Nevertheless, the position estimation process is constrained by the quality of the point cloud and the efficient matching among points (Zaganidis et al. 2018).

In this work, we first describe the error propagation attached to mobile terrestrial LiDAR measurements, and its effects on the estimation of orchard

parameters when only GNSS measurements are considered as localization. Then, we describe an approach which improves the vehicle localization, by fusing a scan matching algorithm with GNSS measurements. For the experimentations, we increment the GNSS uncertainty up to two meters to quantify its influence on orchard parameter estimation. We analyze the case for the following features: crown surface area, crown volume, and porosity. To do so, we implement previously published techniques (Pfeiffer et al. 2018; Trochta et al. 2017; Li et al. 2018) and evaluate their sensitivity to changes in the error of the GNSS antenna, for two cases: (i) GNSS as a solo localization system and (ii) GNSS with scan matching as localization system.

The remaining of the paper is organized as follows: Section 2 describes in detail the experimental setup and the methodology followed in this work. Section 3 presents the experiment developed and the results for the crown parameter estimation. Finally, Section 4 shows the conclusions and future work.

2. Materials and methods

This section describes the experimental setup and the methodology followed to estimate the orchard parameters. First, the methodology describes the error propagation of GNSS over the laser scan measurements; second, it focuses on describing the proposal to overcome such effect.

2.1. Experimental setup

To perform our analysis, we equipped a hydro-pneumatic sprayer with an RTK Leica GPS1200 and a LiDAR Velodyne PUCK; both are assembled in a rigid structure, as shown in Fig. 1.

Figure 1: Hardware used in this work. The platform is equipped with a Velodyne PUCK LiDAR and an RTK Leica GPS1200.

The LiDAR Velodyne PUCK can generate a 3D point cloud (x-y-z positions) of the scanned scene using up to 16 ray readings per frame, with a maximum range of 100 meters and an accuracy of $\pm$ 0.03 meters. The RTK-GNSS system GPS1200+(Leica Geosystems AG, Heerbrugg, Switzerland) provides absolute coordinates and UTC time (synchronized with the LiDAR) with a frequency up to 20 Hz and precision of approximately 20 mm.

2.2. Error propagation

When registering a point cloud using a LiDAR and a GNSS antenna, there are at least three coordinate frames: a global reference frame, $< Global >$; a coordinate frame attached to the GNSS receiver, $< GNSS >$; and a coordinate frame attached to the LiDAR sensor, $< LiDAR >$, as can be seen in Fig. 2.

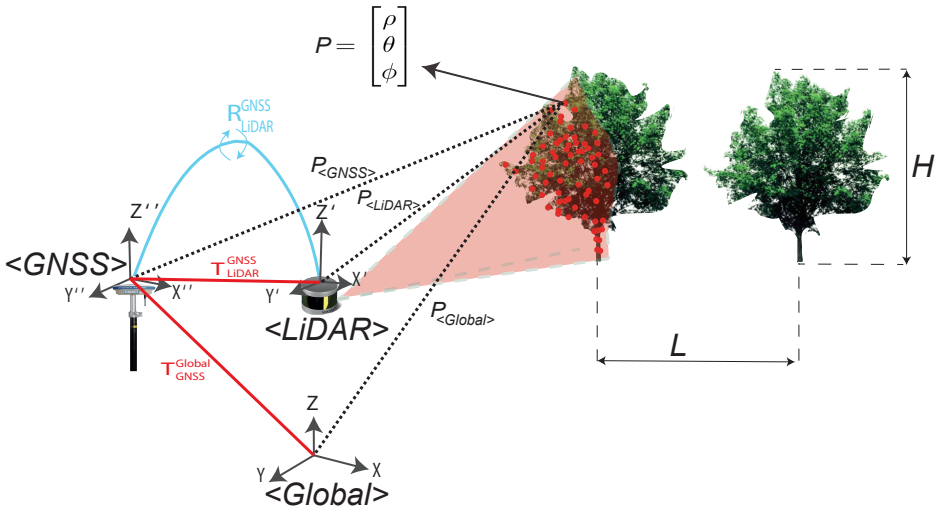

Figure 2: General scheme of the GNSS antenna and the LiDAR used for point cloud acquisition. The point P (in spherical coordinates) acquired in the $< LiDAR >$ coordinate frame is first transformed to the $< GNSS >$ coordinate frame and then to the $< Global >$ reference frame. Twenty trees have been used for this experiment, where the distance between each tree in the same row is L=1 meter and the height of each tree is H=2 meters.

Each point registered from the environment has three coordinates. In our case, such three coordinates are spherical: one range ρ and two angles, θ and ϕ, as shown in Fig. 2 (it is worth to mention that 3D LiDARs might also register a point in cylindrical coordinates, depending on the manufacturer). The point P acquired by the LiDAR in spherical coordinates is then converted

into Cartesian coordinates using the transformation shown in Eq. 1.

$$P_{<LiDAR>} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} \rho \sin\theta \cos\phi \\ \rho \sin\theta \sin\phi \\ \rho \cos\theta \end{bmatrix} \tag{1}$$

It is to be noted that LiDAR readings are not extent of error. If we consider that the spherical coordinates associated with a single point are independent Gaussian random variables, then their covariance matrix will be of the form:

$$\Sigma_{LiDAR} = \begin{bmatrix} \sigma_\rho^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_\phi^2 \end{bmatrix} \tag{2}$$

where σ_ρ^2 is the covariance associated with the LiDAR range reading, and σ_θ^2 and σ_ϕ^2 are the covariances of the associated angles; Σ_{LiDAR} is the covariance matrix associated with a single point $P_{<LiDAR>}$.

The point $P_{<LiDAR>}$ can be transformed into the global coordinate frame according to Eq. 3, considering that the GNSS antenna receives localization information and not orientation (Mezian et al. 2016; Hartzell et al. 2015).

$$\begin{aligned} P_{<Global>} &= P_{<GNSS>} + T_{GNSS}^{Global} \\ P_{<Global>} &= \left(R_{LiDAR}^{GNSS} \cdot (P_{<LiDAR>}) + T_{LiDAR}^{GNSS}\right) + T_{GNSS}^{Global} \end{aligned} \tag{3}$$

To propagate the error, we followed the guidelines presented in (Auat Cheein and Carelli 2013), therefore, analytically we have that:

$$\begin{aligned}\Sigma_{P,Global} &= \Sigma_{LiDAR}^{GNSS} + \Sigma_{GNSS} \\ &= \nabla_P^T \Sigma_{LiDAR} \nabla_P + \Sigma_{GNSS}\end{aligned} \tag{4}$$

where ∇_P is Jacobbian matrix of Eq. 1 and the Jacobbian matrix corresponding to the transformation from the GNSS coordinate frame to the global frame is the identity matrix. It is to be noted that all covariance matrices used here are positive semi-definite.

From the previous statements, we can see that if we have two GNSS positioning systems, each one of them with their corresponding covariance matrix associated with their errors, i.e., Σ_{GNSS}^1 and Σ_{GNSS}^2, and if we previously know that, e.g., $\Sigma_{GNSS}^1 - \Sigma_{GNSS}^2 \succeq 0$, where $\succeq$ stands for *positive semi-definite*, then, for the same point P:

$$(\Sigma_{LiDAR}^{GNSS} + \Sigma_{GNSS}^1) - (\Sigma_{LiDAR}^{GNSS} + \Sigma_{GNSS}^2) \succeq 0.$$

The later means that the higher the covariance associated with the GNSS antenna, the highest the error propagation. Since the above calculation is for a single point P and considering that a point cloud is a concatenation of single points, then the same reasoning can be applied to a covariance matrix associated with a point cloud.

2.3. Overcoming the GNSS positioning error

To overcome the error associated with the GNSS and without the use of external hardware, our work aims to improve the results of the parameters estimation based on the fusion of scan matching estimations and GNSS measurements. The methodology used in this work is depicted in Fig. 3.

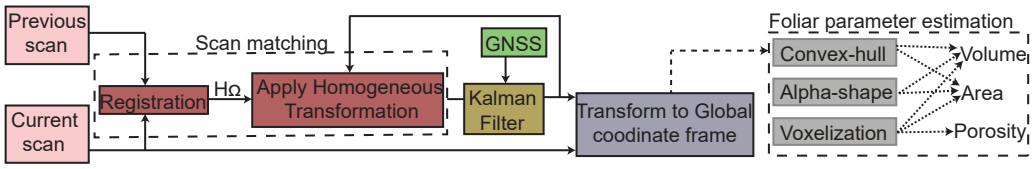

Figure 3: Methodology used for improving the foliar paramater estimation results. The previous and the current laser scans enter in a scan matching algorithm and are fused with the GNSS data. Once all laser scans are transformed to global coordinate frame considering the output of the Kalman Filter, the crown foliar parameters are estimated.

As can be seen from Fig. 3, the first step in our methodology is the scan matching approach. Although several scan matching approaches can be implemented, in this work, the registration process between the previous scan, Λ_k, and the current scan, Ω_k, (where k represents sampling time instant) was accomplished with the use of a 3D ICP algorithm (Besl and McKay 1992). Others can be used, but its analysis is out of the scope of this work. As a result of the registration, the corresponded transformation matrix, H_Ω, was obtained.

The ICP method is an algorithm used to find the rigid transformation H_Ω between a target point cloud Ω_k and a reference point cloud Λ_k so that the matching satisfies a metric criterion.

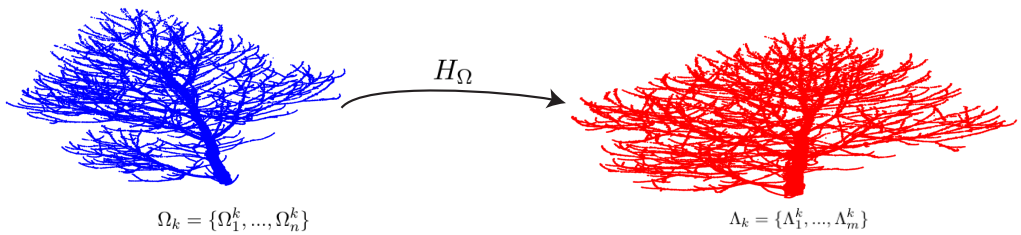

Figure 4: The transformation matrix H_Ω allows to align the point cloud Ω_k with the point cloud Λ_k.

Consider Fig. 4 where the correspondence H_Ω, which aligns point cloud

Ω_k and Λ_k, needs to be found. First, the ICP compute the closest points between the two point clouds as:

$$Y_k = Cp(\Omega_k, \Lambda_k) \tag{5}$$

where Cp represents a closest point operator, therefore, Y_k represents the closest points from Ω_k to Λ_k. To obtain the rigid transformation matrix H_Ω between two sets of points, the ICP minimize the sum of square error between them, as shown below:

$$H_\Omega = \frac{1}{m}\sum_{i=1}^{m} ||\Lambda_i^k - R(Y_i^k) - t||^2 \tag{6}$$

here, t represents translation and R rotation. When applying the alignment to Ω_k the resulting point cloud Ω_{k+1} can be obtained, as seen in Eq. 7.

$$\Omega_{k+1} = H_\Omega(\Omega_k) \tag{7}$$

Then, the distance d between Ω_{k+1} and Λ_k is obtained by:

$$d = \sum_{i=1}^{m} ||\Omega_i^{k+1} - \Lambda_i^k||^2. \tag{8}$$

This process is repeated until d is less than a previous establish value, τ, or when the maximum number of iterations, I_{max}, has been reached. It is worth to mention that this process only warranties the convergence to a local minimum (Yang et al. 2016).

Following the guidelines published in (Manoj et al. 2015), we obtained a close-form covariance of the resultant transformation provided by the ICP

algorithm. This close-form covariance does not make any assumption on the noise present in the sensor data and has no constraints on the estimated rigid transformation. Considering Ω_k and Λ_k, a cost J can be defined as:

$$J = \frac{1}{m}\sum_{j=1}^{m} ||\Lambda_j^k - R(\Omega_j^k) - t||^2. \tag{9}$$

In Eq. 9, R is a rotation matrix and t is a translation vector. Then, the covariance was obtained as follows:

$$cov(H_\Omega) \approx \left(\frac{\partial^2 J}{\partial H_\Omega^2}\right)^{-1} \left(\frac{\partial^2 J}{\partial \digamma \partial H_\Omega}\right) cov(\digamma) \left(\frac{\partial^2 J}{\partial \digamma \partial H_\Omega}\right)^{T} \left(\frac{\partial^2 J}{\partial H_\Omega^2}\right)^{-1} \tag{10}$$

where:

$$\left(\frac{\partial^2 J}{\partial H_\Omega^2}\right) = \begin{bmatrix} \left(\frac{\partial^2 J}{\partial x^2}\right) & \left(\frac{\partial^2 J}{\partial y \partial x}\right) & \left(\frac{\partial^2 J}{\partial z \partial x}\right) & \left(\frac{\partial^2 J}{\partial a \partial x}\right) & \left(\frac{\partial^2 J}{\partial b \partial x}\right) & \left(\frac{\partial^2 J}{\partial c \partial x}\right) \\ \left(\frac{\partial^2 J}{\partial x \partial y}\right) & \left(\frac{\partial^2 J}{\partial y^2}\right) & \left(\frac{\partial^2 J}{\partial z \partial y}\right) & \left(\frac{\partial^2 J}{\partial a \partial y}\right) & \left(\frac{\partial^2 J}{\partial b \partial y}\right) & \left(\frac{\partial^2 J}{\partial c \partial y}\right) \\ \left(\frac{\partial^2 J}{\partial x \partial z}\right) & \left(\frac{\partial^2 J}{\partial y \partial z}\right) & \left(\frac{\partial^2 J}{\partial z^2}\right) & \left(\frac{\partial^2 J}{\partial a \partial z}\right) & \left(\frac{\partial^2 J}{\partial b \partial z}\right) & \left(\frac{\partial^2 J}{\partial c \partial z}\right) \\ \left(\frac{\partial^2 J}{\partial x \partial a}\right) & \left(\frac{\partial^2 J}{\partial y \partial a}\right) & \left(\frac{\partial^2 J}{\partial z \partial a}\right) & \left(\frac{\partial^2 J}{\partial a^2}\right) & \left(\frac{\partial^2 J}{\partial b \partial a}\right) & \left(\frac{\partial^2 J}{\partial c \partial a}\right) \\ \left(\frac{\partial^2 J}{\partial x \partial b}\right) & \left(\frac{\partial^2 J}{\partial y \partial b}\right) & \left(\frac{\partial^2 J}{\partial z \partial b}\right) & \left(\frac{\partial^2 J}{\partial a \partial b}\right) & \left(\frac{\partial^2 J}{\partial b^2}\right) & \left(\frac{\partial^2 J}{\partial c \partial b}\right) \\ \left(\frac{\partial^2 J}{\partial x \partial c}\right) & \left(\frac{\partial^2 J}{\partial y \partial c}\right) & \left(\frac{\partial^2 J}{\partial z \partial c}\right) & \left(\frac{\partial^2 J}{\partial a \partial c}\right) & \left(\frac{\partial^2 J}{\partial b \partial c}\right) & \left(\frac{\partial^2 J}{\partial c^2}\right) \end{bmatrix}_{6\times 6}$$

$$\left(\frac{\partial^2 J}{\partial \digamma \partial H_\Omega}\right) = \begin{bmatrix} \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial x} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial x} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial x} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial x} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial x} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial x} \\ \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial y} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial y} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial y} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial y} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial y} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial y} \\ \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial z} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial z} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial z} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial z} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial z} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial z} \\ \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial a} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial a} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial a} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial a} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial a} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial a} \\ \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial b} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial b} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial b} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial b} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial b} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial b} \\ \frac{\partial^2 J}{\partial \Omega_{ix}^k \partial c} & \frac{\partial^2 J}{\partial \Omega_{iy}^k \partial c} & \frac{\partial^2 J}{\partial \Omega_{iz}^k \partial c} & \frac{\partial^2 J}{\partial \Lambda_{ix}^k \partial c} & \frac{\partial^2 J}{\partial \Lambda_{iy}^k \partial c} & \frac{\partial^2 J}{\partial \Lambda_{iz}^k \partial c} \end{bmatrix}_{6\times 6n}.$$

Here, (x, y, z) represents the translations of H_Ω in the x, y and z coordinates, respectively, and (a, b, c) represents the rotations of H_Ω in yaw, pitch, and roll, respectively. Additionally, n is the number of correspondences between the point clouds Λ_i^k and Ω_i^k. On the other hand, $cov(\digamma)$ represents the noise present in the (x, y, z) components acquired by sensor readings, as shown in Eq. 11.

$$cov(\digamma) = \begin{bmatrix} cov(\Lambda_1^k) & 0 & ... & 0 \\ 0 & cov(\Omega_1^k) & ... & 0 \\ 0 & \vdots & \ddots & 0 \\ 0 & 0 & ... & cov(\Omega_n^k) \end{bmatrix} \tag{11}$$

The above expressions were used for information fusion purposes for getting a more accurate position estimation. This fusion was done via a Kalman filter fusing the data associated with the GNSS and the scan matching. It is worth mentioning that this algorithm must be initialized, with prior knowledge of the position. For the objective of improving the orchard parameter

estimation only with remote measurements, the process model of the Kalman filter was set as an identity matrix. More information about the approach implemented in this work can be found in (Sun and Deng 2004; Caron et al. 2006). The sensor fusion procedure for improving localization is shown in Algorithm 1.

In Algorithm 1, lines of code (1) to (3) use prior knowledge for initialization; lines of code (9) to (17) show the ICP scan matching approach with the information provided in lines of code (7) to (9); in lines of code (20) to (26), the procedure for obtaining ICP covariance is computed. The measurement matrix is defined in line (27) and the covariance matrix is defined in line (28). The estimated position, $\hat{X}_k$, is obtained with a Kalman Filter (line 29) considering the created measurement and covariance matrices. Line (30) transform Ω_k to global coordinate system according to the estimated position $\hat{X}_k$. Finally, the transformed point cloud, S_k, is stored in $\mathbf{M}$ (line 32) to create an environment point cloud.

Once the environment point cloud, $\mathbf{M}$, is obtained, we performed the foliar parameter estimation. For this manuscript, we have selected the following orchard features: crown volume, crown surface area and porosity, since they are the most common features extracted from point clouds. In addition, we have implemented, as features extraction procedure, the following computational approaches: convex-hull, for crown volume and crown surface area; alpha-shape, also for crown volume and crown surface area; and voxelization, for crown volume, crown surface area and porosity, as presented in (Pfeiffer et al. 2018).

Algorithm 1: Sensor fusion using scan matching and GNSS readings.

input : [$GNSS_k$ data and LiDAR scan Ω_k]
output : [Environment point cloud $\boldsymbol{M}$]

Define Covariance R_1 of $GNSS_k$;
Define $cov(F)$ according to Eq. 11;
Define Kalman parameters;
Obtain Ω_1;
for $k \leftarrow 2$ **to** *final position* **do** `// For each position`
  $z_1 \leftarrow GNSS_k$;
  Get Ω_k;
  $\Lambda = \Omega_{k-1}$;
  while $i < I_{max}$ **do** `// Calculate Correspondance`
    $Y = Cp(\Omega_i, \Lambda)$;
    Find H_Ω that minimize Eq.6;
    $\Omega_{i+1} \leftarrow H_\Omega(\Omega_i)$;
    Calculate the distance d according to Eq.8;
    if $d < \tau$ **then**
      stop Iteration i;
      break;
    end
  end
  $z_2 \leftarrow H_\Omega(\Omega_k)$;
  for $j \leftarrow 1$ **to** m **do** `// Calculate covariance`
    E$\leftarrow$ Evaluate $\left(\frac{\partial^2 J}{\partial H_\Omega^2}\right)$;
    $\left(\frac{\partial^2 J}{\partial H_\Omega^2}\right)_j \leftarrow \left(\frac{\partial^2 J}{\partial H_\Omega^2}\right)_{j-1}$ + E;
    V$\leftarrow$Evaluate $\left(\frac{\partial^2 J}{\partial F\, \partial H_\Omega}\right)$;
    $\left(\frac{\partial^2 J}{\partial F\, \partial H_\Omega}\right)_j \leftarrow [\ \left(\frac{\partial^2 J}{\partial F\, \partial H_\Omega}\right)_{j-1}$ V] ;
  end
  $R_2 \leftarrow cov(H_\Omega)$ according Eq.10;
  Create measurement matrix $Z = [z_1; z_2]$;
  Create covariance matrix $R = [R_1; R_2]$;
  $\hat{X}_k \leftarrow$ Kalman Filter(Z, R);
  $S_k \leftarrow$ Transform Ω_k to global coordinate system according to $\hat{X}_k$;
  $\mathbf{M} \leftarrow [\mathbf{M}\ \ S_k]$;
end

3. Experimental results

The tests were carried out at an apple tree grove. The platform traversed approximately 20 meters back and forward through an apple tree corridor. The sampling time of the RTK was set to 0.05 seconds, synchronized with the sampling time of the LiDAR; 150 trials were performed. Since the RTK has an absolute error of 0.01 meters (horizontally), it was used as ground truth. Then, to test the sensitivity of the grove features (i.e., crown surface area, crown volume and porosity) to the GNSS antenna error, artificial Gaussian noise was added to the GNSS positioning until reaching a maximum of 2 meters of absolute error (which is consistent with low-cost GNSS antennas), for each trial, and repeated ten times. Figure 5 shows part of the experimental data used in this work, where the LiDAR was positioned in several parts from the environment to register a point cloud. Figure 5a shows a partial view of a tree (green points) reconstructed with a LiDAR and GNSS antenna with 0.01 meters of absolute error in axis x and y. The same point cloud is then reconstructed using a GNSS antenna with a positioning error of 2 meters, as shown in Fig. 5d. As can be seen, as the absolute error increases, the point cloud becomes more disperse; in fact, several stems appear duplicated.

3.1. Propagation of GNSS error in crown parameter estimation

Figure 6-8 shows the results obtained during the trials. The three methodologies: voxelization, convex-hull and alpha-shape, were used to estimate the crown volume and crown surface area. However, to estimate porosity, only voxelization was used (Béland et al. 2014). These figures show the mean

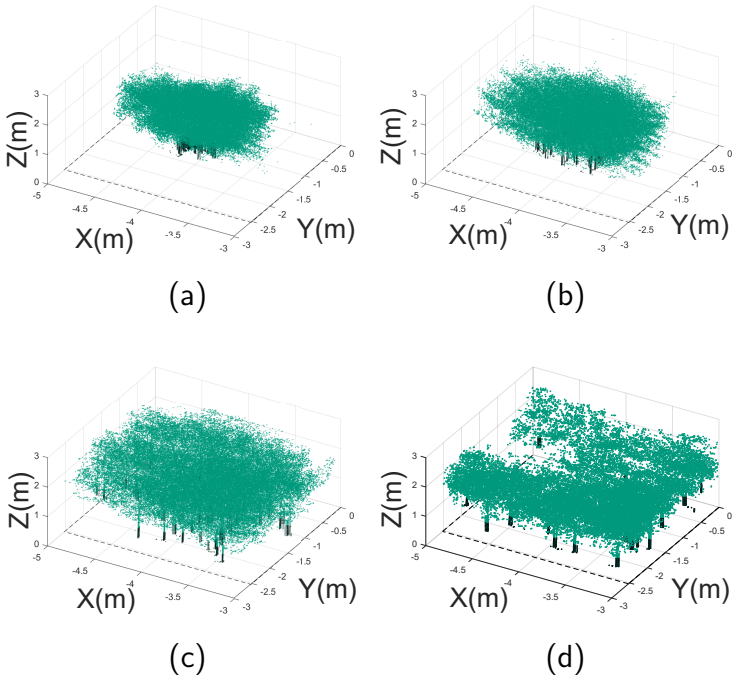

Figure 5: The ground-truth and three point clouds with different positioning error associated with the GNSS antenna. The GNSS error was established in 0.01, 0.2, 1 and 2 meters for Figure 5a-Fig. 5d, respectively.

in solid magenta, the standard deviations in dashed dark line and single estimations in dotted dark line.

From Figure 6-8, a number of lessons are learned:

- Regarding crown volume estimation, as the positioning error in the GNSS antenna increases up to 0.8 meters approximately, the estimated volume also increases for the three approaches (see Figs. 6a-6c). For the convex-hull and alpha-shape case, the volume estimation experiences a saturation behavior after one meter of error. Thus, if we increase the error in the positioning system, the volume estimation does not increase.

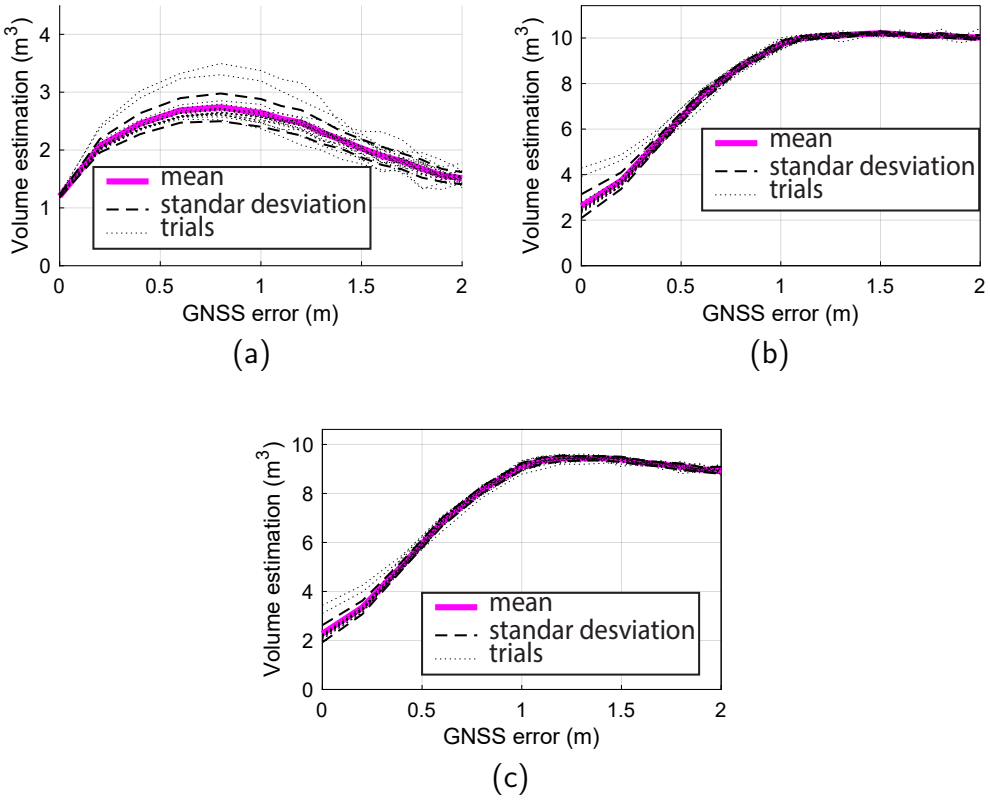

Figure 6: Experimental results for crown volume estimation. Figures 6a-6c shows the results for voxelization, convex-hull and alpha-shape, respectively. In all figures, the solid magenta line represents the mean estimation, whereas the dashed line is the standard deviation, and the dotted dark line are single estimations for each trial.

- When using voxelization, the estimated volume presented in Fig. 6a shows an opposite behavior compared to the other two approaches: it decays after 0.8 meters in the positioning error. This reason of this outcome is that as the error in the positioning increases, the cloud point becomes more disperse, as shown in Fig. 5. Therefore, the size of the voxels is smaller and the total estimated volume decreases.

- Regarding crown surface area estimation, it can be seen from Figs. 7a-

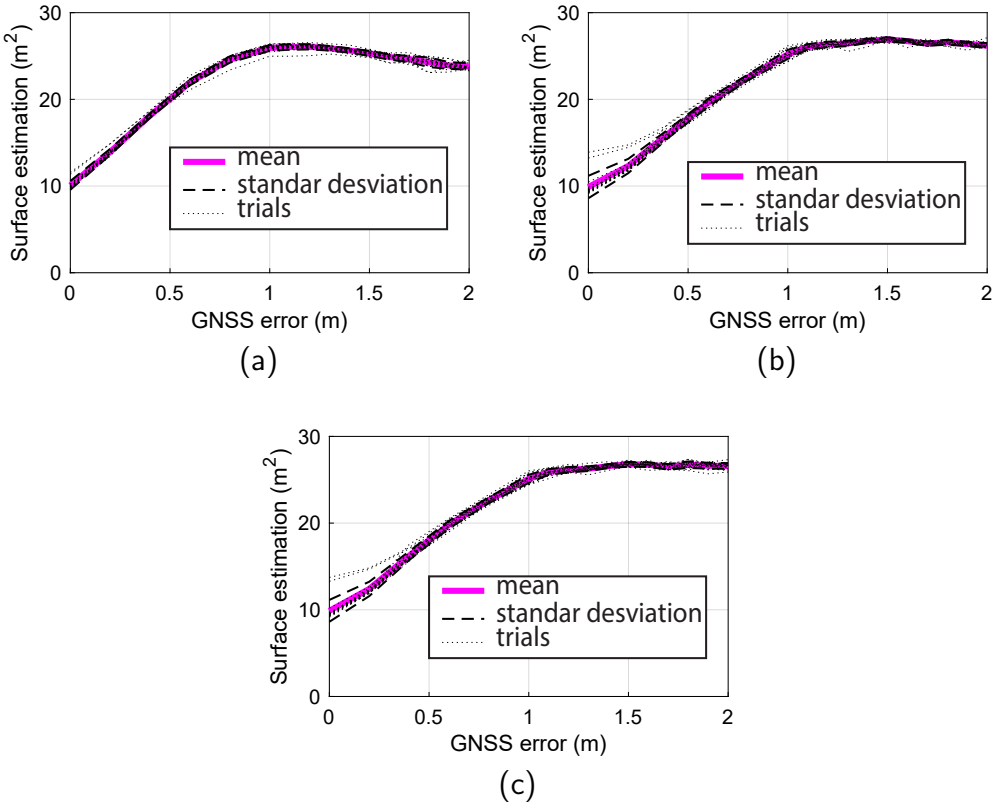

Figure 7: Experimental results for crown surface area estimation. Figures 7a-7c show the results for voxelization, convex-hull and alpha-shape, respectively. In all figures, the solid magenta line represents the mean estimation, whereas the dashed line is the standard deviation, and the dotted dark line are single estimations for each trial.

7c that the three approaches behave in a similar fashion: the estimation increases until approximately one meter of positioning error and then it remains almost constant. The voxelization approach, however, shows a slight decrease of approximately 10% with respect to the maximum observed. The latter is also associated with the fact that as the error in the position increases, the point cloud becomes more disperse and voxels smaller.

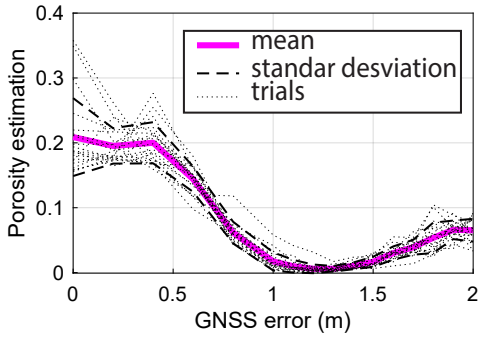

Figure 8: Porosity results using voxelization. The solid magenta line represents the mean estimation, whereas the dashed line is the standard deviation, and the dotted dark line are single estimations for each trial.

- Regarding porosity, from the three methods, only voxelization allows for its estimation. As it can be seen, as the positioning error increases, porosity drastically decreases, which is consistent with porosity definition presented in (Béland et al. 2014; Pfeiffer et al. 2018): as the points become more dispersed, less shadow is expected.

3.2. GNSS and scan matching as a localization system

The estimated position obtained from the Kalman filter was used to transform the point cloud data to global coordinates. Once all point clouds were transformed, the orchard parameter estimation was obtained following the same methodology described in Section 3.1. Then, the results of both the original approach and the proposal are compared using the mean square error.

3.2.1. Orchard parameter estimation

The results showed in Figure 9-10, follow the same behavior of the results showed in Figure 6-7, where the tendency of crown volume and the crown

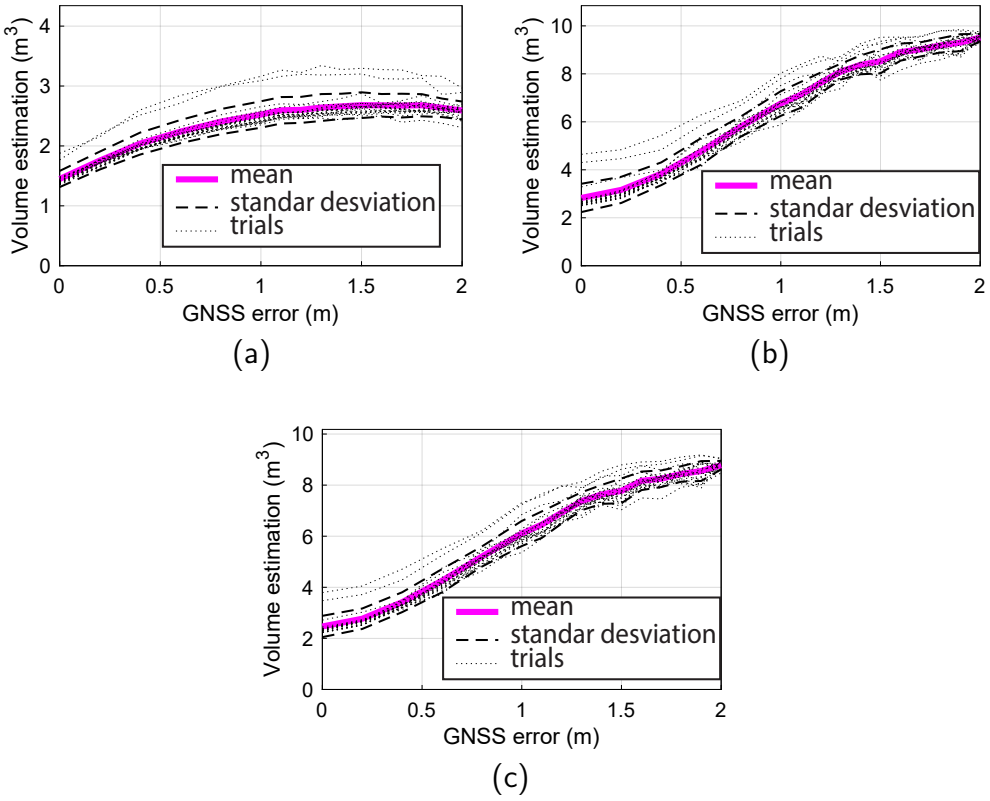

Figure 9: Experimental results. Figures 9a-9c show the crown volume estimated with the three different methods implemented in this work: voxelization, convex-hull and alpha-shape, respectively.

surface area is to increase. Based on the results obtained, the maximum is reached when GNNS error is equal to 2 meters, quite the contrary to the results from Figure 6-7 where the maximum is reached when GNSS error is equal to one meter. Therefore, the results obtained in this case give a more accurate estimation of orchard parameters: for crown surface area and crown volume estimation, for the three methodologies (voxelization, convex-hull and alpha-shape).

Additionally, Fig. 11 shows the results after evaluating porosity. As can

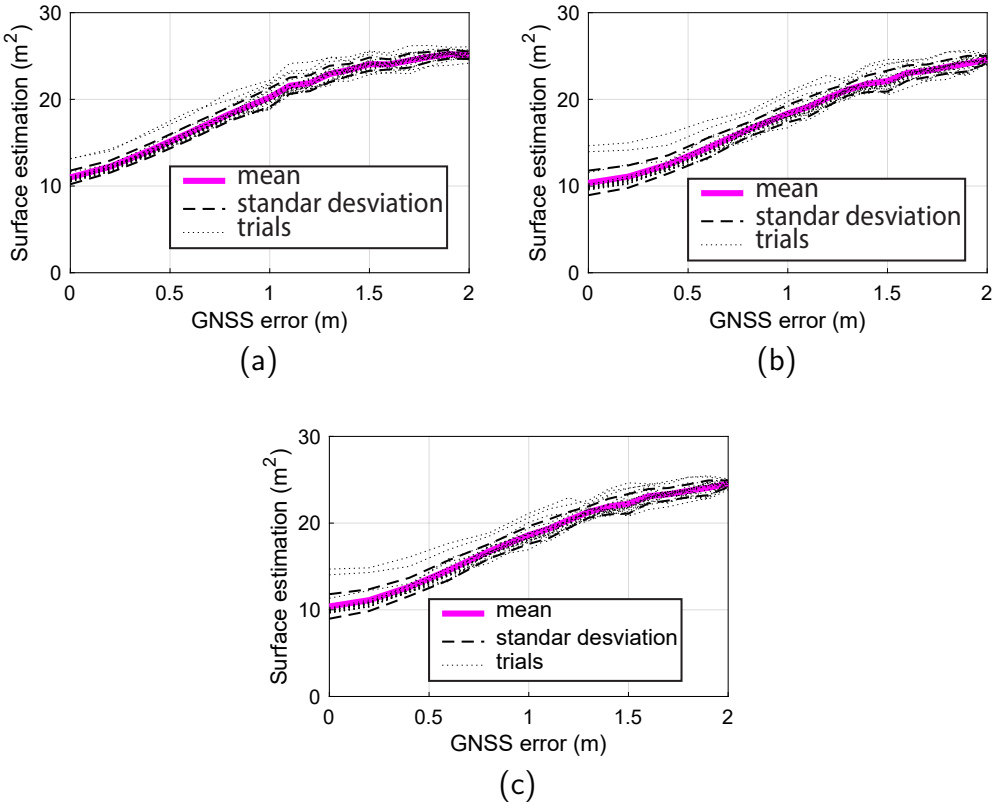

Figure 10: Figures 10a-10c show the results for crown surface area estimation. The solid magenta line represents the mean estimation, whereas the dashed line is the standard deviation, and the dotted dark line are single estimations for each trial.

be seen, when compared to Fig. 8, the decay of the porosity estimation as the GNSS error increases is smoother. The latter implies that the point cloud is less sparse when using the localization fusion approach to overcome the error in the GNSS antenna.

3.2.2. Comparison analysis

Figure 12-13 shows a comparison of the mean square error between the results obtained with GNSS only and its fusion with scan matching for or-

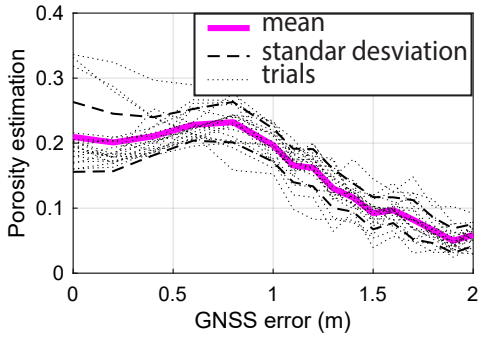

Figure 11: Porosity results using voxelization. The solid magenta line represents the mean estimation, whereas the dashed line is the standard deviation, and the dotted dark line are single estimations for each trial.

chard parameters estimation. As can be seen, there is a reduction in the error associated with the estimation of all the features: crown surface area, crown volume, and porosity. The convex-hull results for crown surface area and volume are shown in Figure 12a and Figure 12b, respectively. In both cases, there is a notorious reduction of the error percentage. For example, in one meter of GNSS error, the results show that considering only GNSS as a localization system, the error increases up to 158 %, and when fused with the scan matching, the error increases up to 79%. Thus, an improvement of almost 50% was reached. The alpha-shape results show an improvement in the error associated with the estimation, as can be seen in Figures 12c and 12d. The estimation errors decrease in at least 20% for all cases. The voxelization approach presents the results for crown surface area, crown volume and porosity in Figures 13a, 13b and 13c, respectively. From these results, it can be noted that the estimation error of the crown surface area and porosity has the same tendency as the previous methods. However, in the volume case, the presented approach seems to be worse than the GNSS

trial after 1.2 meters of error. This outcome was obtained because the number of voxels within the workspace begins to decrease when exceeding 0.8 meters of GNSS error. Although such value is closer to the real value, it does not represent a consistent measure since the reconstructed point cloud is extremely dispersed. Even though our approach reduces the estimation error, it is subjected to similar effects to those of only the GNSS approach.

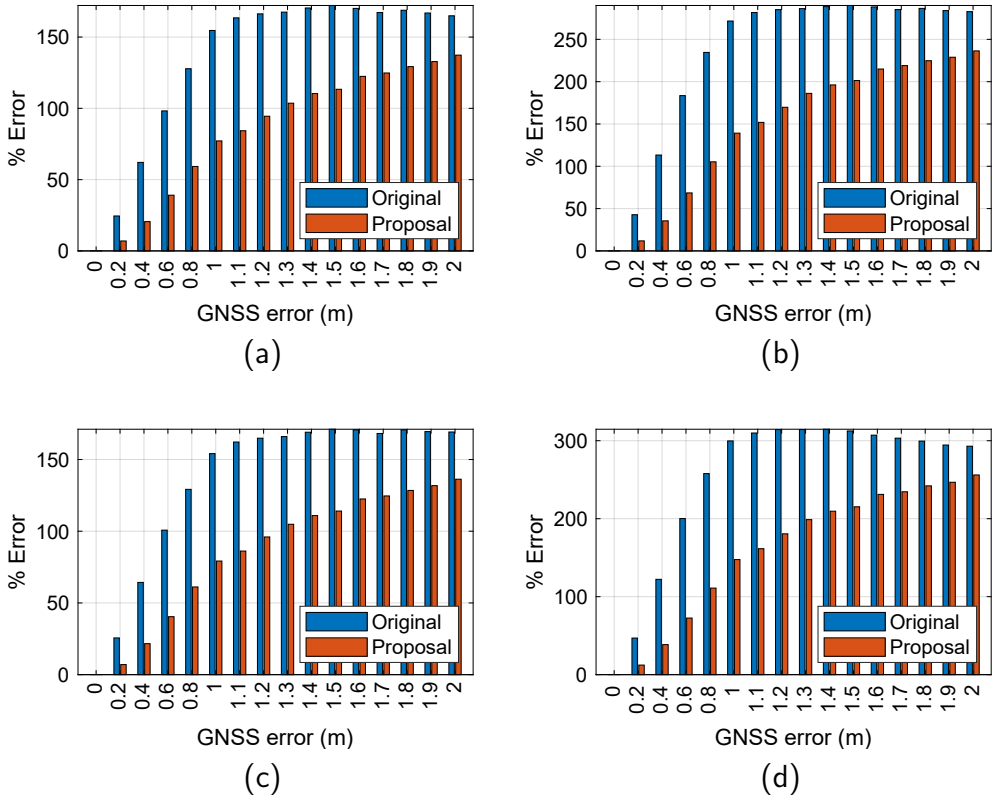

Figure 12: Mean square error percentage for the original (GNSS only) and the proposed (GNSS with scan matching) approach. The convex-hull results for crown surface area Figures 12a and volume 12b; Alpha-shape surface area and volume in Figures 12c and 12d. The blue and the red bar represent the error percentage in the volume estimation process when only GNSS is consider and when GNSS is fused with scan matching measurements, respectively.

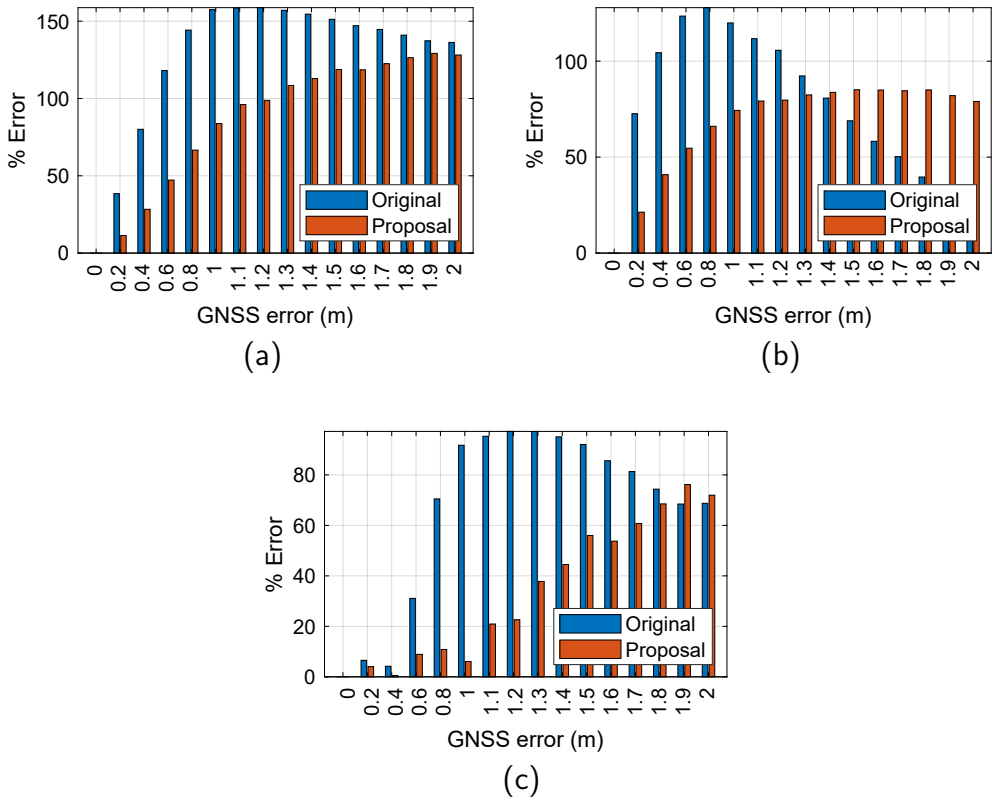

Figure 13: Mean square error percentage for the original (GNSS only) and the proposed (GNSS with scan matching) approach. Figures 13a-13b show the voxelization approach mean square error for crown surface area and volume, respectively. Figure 13c shows the voxelization approach mean square error for porosity.

4. Conclusion

This work has analyzed the sensitivity of orchard features obtained via LiDAR, to GNSS positioning error. In particular, the crown volume, the crown surface area, and the porosity were studied, using three different computational approaches: convex-hull, alpha-shape, and voxelization. The experimentation was carried out at an apple grove. The analysis started with a positioning error of 0.01 meters provided by an RTK and then it was arti-

ficially increased, adding Gaussian random noise, reaching up to two meters of error, as in low-cost GNSS receivers. The results considering only GNSS measurements have shown that crown surface area and crown volume experienced an increment as the positioning error grew up to one meter –for convex-hull and alpha-shape cases–, and then they remained almost stationary. For the voxelization case, crown surface area estimate behaved as in the previous approaches, but crown volume estimate decreased due to the reduction in the voxel size. In the case of porosity estimation, it decreased until positioning error reached one meter, and then started to grow again. To improve the above results, based only only on the LiDAR data available –thus avoiding the need of extra hardware– a scan matching approach (specifically, the ICP) was implemented and fused with the GNSS measurements, in order to decrease the localization error. The results showed a notorious improvement against the original implementation. Errors in the crown surface area, crown volume and porosity estimation reduced up to 20%, by adding the ICP when the GNSS error was of 1.2 meters and up to 50% for smaller errors. These outcomes suggest that the proposed approach can improve the results of foliar parameters estimation, therefore, leading to better decisions in agricultural operations without an increment in costs of external hardware.

Acknowledgement(s)

The authors would like to thank to CONICYT FB0008, CONICYT FONDECYT 1171431, PIIC 030/2018 and DGIIP-UTFSM Chile for partially funding this research.

References

Auat Cheein, F. A. and R. Carelli (2013). Agricultural robotics: Unmanned robotic service units in agricultural tasks. *IEEE Industrial Electronics Magazine 7*(3), 48–58.

Auat Cheein, F. A., J. Guivant, R. Sanz, A. Escolà, F. Yandún, M. Torres-Torriti, and J. R. Rosell-Polo (2015). Real-time approaches for characterization of fully and partially scanned canopies in groves. *Computers and Electronics in Agriculture 118*, 361–371.

Béland, M., D. D. Baldocchi, J. L. Widlowski, R. A. Fournier, and M. M. Verstraete (2014). On seeing the wood from the leaves and the role of voxel size in determining leaf area distribution of forests with terrestrial LiDAR. *Agricultural and Forest Meteorology 184*, 82–97.

Besl, P. J. and N. D. McKay (1992). A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence 14*(2), 239–256.

Boughorbel, F., A. Koschan, B. Abidi, and M. Abidi (2004). Gaussian fields: A new criterion for 3D rigid registration. *Pattern Recognition 37*(7), 1567–1571.

Caron, F., E. Duflos, D. Pomorski, and P. Vanheeghe (2006). GPS/IMU data fusion using multisensor Kalman filtering: Introduction of contextual aspects. *Information Fusion 7*(2), 221–230.

Donoso, F. A., K. J. Austin, and P. R. McAree (2017). How do ICP variants perform when used for scan matching terrain point clouds? *Robotics and Autonomous Systems 87*, 147–161.

Gaulton, R. and T. J. Malthus (2010). LiDAR mapping of canopy gaps in continuous cover forests: A comparison of canopy height model and point cloud based techniques. *International Journal of Remote Sensing 31*(5), 1193–1211.

Glennie, C. (2008). Rigorous 3D error analysis of kinematic scanning LIDAR systems. *Journal of Applied Geodesy 1*(3), 147–157.

Goulden, T. and C. Hopkinson (2010). The forward propagation of integrated system component errors within airborne lidar data. *Photogrammetric Engineering and Remote Sensing 76*(5), 589–601.

Granger, S. and X. Pennec (2002). Multi-scale EM-ICP: A fast and robust approach for surface registration. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Volume 2353, pp. 418–432.

Grant, W. S., R. C. Voorhies, and L. Itti (2019). Efficient Velodyne SLAM with point and plane features. *Autonomous Robots 43*(5), 1207–1224.

Hartzell, P. J., P. J. Gadomski, C. L. Glennie, D. C. Finnegan, and J. S. Deems (2015). Rigorous error propagation for terrestrial laser scanning with application to snow volume uncertainty. *Journal of Glaciology 61*(230), 1147–1158.

Kim, P., J. Chen, and Y. K. Cho (2018). SLAM-driven robotic mapping and registration of 3D point clouds. *Automation in Construction 89*(December 2017), 38–48.

Lamine Tazir, M., T. Gokhool, P. Checchin, L. Malaterre, and L. Trassoudaine (2018). CICP: Cluster Iterative Closest Point for sparse–dense point cloud registration. *Robotics and Autonomous Systems 108*, 66–86.

Li, L., X. He, J. Song, Y. Liu, A. Zeng, L. Yang, C. Liu, and Z. Liu (2018). Design and experiment of variable rate orchard sprayer based on laser scanning sensor. *International Journal of Agricultural and Biological Engineering 11*(1), 101–108.

Lin, W., Y. Meng, Z. Qiu, S. Zhang, and J. Wu (2017). Measurement and calculation of crown projection area and crown volume of individual trees based on 3D laser-scanned point-cloud data. *International Journal of Remote Sensing 38*(4), 1083–1100.

Magnusson, M., A. Nüchter, C. Lörken, A. J. Lilienthal, and J. Hertzberg (2009). Evaluation of 3D registration reliability and speed-A comparison of ICP and NDT. In *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3907–3912.

384 Malavazi, F. B., R. Guyonneau, J. B. Fasquel, S. Lagrange, and F. Mercier (2018). LiDAR-
385 only based navigation algorithm for an autonomous agricultural robot. *Computers and*
386 *Electronics in Agriculture 154*(February), 71–79.

387 Manoj, P. S., L. Bingbing, Y. Rui, and W. Lin (2015). A closed-form estimate of 3D
388 ICP covariance. In *Proceedings of the 14th IAPR International Conference on Machine*
389 *Vision Applications, MVA 2015*, Number 3, pp. 526–529.

390 Mezian, M., B. Vallet, B. Soheilian, and N. Paparoditis (2016). Uncertainty propagation
391 for terrestrial mobile laser scanner. In *International Archives of the Photogrammetry,*
392 *Remote Sensing and Spatial Information Sciences - ISPRS Archives*, Volume 41, pp.
393 331–335.

394 Palleja, T., M. Tresanchez, M. Teixido, R. Sanz, J. R. Rosell, and J. Palacin (2010).
395 Sensitivity of tree volume measurement to trajectory errors from a terrestrial LIDAR
396 scanner. *Agricultural and Forest Meteorology 150*(11), 1420–1427.

397 Peng, C., P. Roy, J. Luby, and V. Isler (2016). Semantic Mapping of Orchards. *IFAC-*
398 *PapersOnLine 49*(16), 85–89.

399 Pfeiffer, S. A., J. Guevara, F. A. Cheein, and R. Sanz (2018, mar). Mechatronic terrestrial
400 LiDAR for canopy porosity and crown surface estimation. *Computers and Electronics*
401 *in Agriculture 146*, 104–113.

402 Ren, R., H. Fu, and M. Wu (2019). Large-scale outdoor slam based on 2d lidar. *Electronics*
403 *(Switzerland) 8*(6).

404 Rilling, S., M. Nielsen, A. Milella, C. Jestel, P. Frohlich, and G. Reina (2017). A multisen-
405 sor platform for comprehensive detection of crop status: Results from two case studies.
406 *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveil-*
407 *lance, AVSS 2017*.

408 Rosell, J. R., J. Llorens, R. Sanz, J. Arnó, M. Ribes-Dasi, J. Masip, A. Escolà, F. Camp,
409 F. Solanelles, F. Gràcia, E. Gil, L. Val, S. Planas, and J. Palacín (2009). Obtaining

410 the three-dimensional structure of tree orchards from remote 2D terrestrial LIDAR
411 scanning. *Agricultural and Forest Meteorology 149*(9), 1505–1515.

412 Rosell, J. R. and R. Sanz (2012). A review of methods and applications of the geometric
413 characterization of tree crops in agricultural activities. *Computers and Electronics in*
414 *Agriculture 81*, 124–141.

415 Sanz, R., J. Llorens, A. Escolà, J. Arnó, S. Planas, C. Román, and J. R. Rosell-Polo (2018).
416 LIDAR and non-LIDAR-based canopy parameters to estimate the leaf area in fruit trees
417 and vineyard. *Agricultural and Forest Meteorology 260-261*(October), 229–239.

418 Segal, A. V., D. Haehnel, and S. Thrun (2009). Generalized-ICP. *Robotics: Science and*
419 *Systems 2*(4), 435.

420 St-Onge, B., C. Vega, R. A. Fournier, and Y. Hu (2008). Mapping canopy height using
421 a combination of digital stereo-photogrammetry and lidar. *International Journal of*
422 *Remote Sensing 29*(11), 3343–3364.

423 Sun, S. L. and Z. L. Deng (2004). Multi-sensor optimal information fusion Kalman filter.
424 *Automatica 40*(6), 1017–1023.

425 Trochta, J., M. Kruček, T. Vrška, and K. Kraâl (2017). 3D Forest: An application
426 for descriptions of three-dimensional forest structures using terrestrial LiDAR. *PLoS*
427 *ONE 12*(5), 1–17.

428 Underwood, J., A. Wendel, B. Schofield, L. McMurray, and R. Kimber (2017). Efficient
429 in-field plant phenomics for row-crops with an autonomous ground vehicle. *Journal of*
430 *Field Robotics 34*(6), 1061–1083.

431 Yang, J., H. Li, D. Campbell, and Y. Jia (2016). Go-ICP: A Globally Optimal Solution to
432 3D ICP Point-Set Registration. *IEEE Transactions on Pattern Analysis and Machine*
433 *Intelligence 38*(11), 2241–2254.

Zaganidis, A., L. Sun, T. Duckett, and G. Cielniak (2018). Integrating Deep Semantic Segmentation into 3-D Point Cloud Registration. *IEEE Robotics and Automation Letters 3*(4), 2942–2949.